In the Claims:

1. (withdrawn) (previously presented) A method of executing a processor instruction, said method comprising:

fetching from memory a first machine language instruction comprising an instruction segment;

responsive to a trigger pattern in said first machine language instruction, modifying said instruction segment to form a second machine language instruction; and

executing on said processor said second machine language instruction.

2. (withdrawn) (original) The method of Claim 1 wherein said modifying substitutes a bit pattern of a subset of said instruction segment.

3. (withdrawn) (original) The method of Claim 2 further comprising repeating said fetching, said modifying and said executing without executing an intervening machine language instruction to change an instruction modification information value utilized by said modifying.

4. (withdrawn) (original) The method of Claim 1 wherein said executing comprises executing microcode.

5. (withdrawn) (original) The method of Claim 1 wherein said trigger pattern is associated with a particular execution unit of said processor.

6. (withdrawn) (original) The method of Claim 1 wherein said first machine language instruction comprises a very long instruction word.

7. (withdrawn) (original) The method of Claim 6 wherein said instruction segment comprises an atomic portion of said very long instruction word.

8. (currently amended) A method of executing a processor instruction, said method comprising:

fetching from memory a first machine language instruction comprising an instruction segment;

responsive to a trigger pattern in said first machine language instruction, accessing instruction modification information from a memory;

modifying said instruction segment according to said instruction modification information and information associated with said trigger pattern to form a second machine language instruction; ~~and~~

executing on said processor said second machine language instruction; and

wherein said memory comprises a plurality of entries, each entry storing instruction modification information.

9. (cancelled)

10. (currently amended) The method of Claim [[9]] 8 further comprising advancing a queue structure to a next entry storing instruction modification information in said memory.

11. (original)The method of Claim 10 wherein said advancing comprises advancing a pointer to indicate said next entry storing instruction modification information in said memory.

12. (original)The method of Claim 10 further comprising repeating said fetching, accessing, modifying and executing without executing an intervening machine language instruction to change any of said plurality of instruction modification information.

13. (original)The method of Claim 8 wherein said executing comprises executing microcode.

14. (original)The method of Claim 8 wherein said trigger pattern is associated with a particular execution unit of said processor.

15. (original)The method of Claim 8 wherein said first machine language instruction comprises a very long instruction word that comprises a plurality of instruction segments.

16. (original)The method of Claim 15 wherein said instruction segment comprises an atomic portion of said very long instruction word.

17. (currently amended) A computer system comprising:

a memory for storing a first machine language instruction;

a second memory for storing a plurality of instruction modification information;

a processor coupled to said memory for executing machine language instructions;

said processor also for implementing a method, said method comprising:

fetching from said memory said first machine language instruction comprising an instruction segment from said memory;

responsive to a trigger pattern in said first machine language instruction, modifying said instruction segment using said

instruction modification information to form a second machine language instruction; and

executing on said processor said second machine language instruction.

18. (original)The computer system of Claim 17 further comprising a cache for caching said first machine language instruction.

19. (original)The computer system of Claim 18 wherein said processor pipelines instruction execution.

20. (cancelled)

21. (original)The computer system of Claim [[20]] 17 wherein said second memory comprises a queue.

22. (original)The computer system of Claim [[20]] 21 wherein said modifying comprises accessing an instruction modification information from said second memory and modifying said instruction segment according to said instruction modification information and information associated with said trigger pattern to form said second machine language instruction

23. (currently amended) A memory stored packet contained within a very long instruction word, said packet comprising:

a trigger pattern to initiate modification of a segment of said very long instruction word;

a first field to indicate a portion of said segment to be modified;

a bit to indicate that a queue of instruction modification data is to be advanced in association with modification of said segment to be modified; and

a second field to indicate how to modify said portion of said segment.

24. (original)The packet of Claim 23 wherein said second field indicates a number of bits of instruction modification information to be substituted into said segment to be modified.

25. (original)The packet of Claim 23 wherein said second field is operable to indicate substitution of a single bit into said segment to be modified.

26. (cancelled)

27. (original)The packet of claim 23 wherein said trigger pattern identifies said segment according to a type of said segment.

28. (original)The packet of claim 27 wherein said trigger pattern identifies an arithmetic logic unit segment.

29. (original)The packet of claim 27 wherein said trigger pattern identifies a floating point logic unit segment.

30. (original)The packet of claim 27 wherein said trigger pattern identifies a memory unit segment.

31. (original)The packet of Claim 27 wherein said trigger pattern identifies a branch unit segment.

32. (original)The packet of Claim 23 wherein said trigger pattern identifies said segment according to a position of said segment in said very long instruction word.

33. (currently amended) A method of modifying a machine language instruction, said method comprising:

accessing said machine language instruction from memory;

recognizing a trigger pattern in said machine language instruction;

identifying a portion of said machine language instruction; and

modifying said portion of said machine language instruction from a queue of instruction modifications to form a second machine language instruction.

34. (original)The method of Claim 33 wherein said identifying comprises decoding said trigger pattern to identify said portion of said machine language instruction.

35. (original)The method of Claim 34 wherein said portion of said machine language instruction is identified according to a type of said portion.

36. (original)The method of Claim 34 wherein said portion of said machine language instruction is identified according to a location of said portion within said machine language instruction.

37. (previously presented)A method of executing an instruction word of a processor comprising:

accessing from memory an instruction word comprising a plurality of instruction segments and a trigger portion;

based on said trigger portion, identifying a portion of information of a memory queue for selection thereof;

based on said trigger portion, identifying a portion of one of said plurality of instruction segments;

modifying said portion of said one of said plurality of instruction segments with said portion of information of said memory queue; and

dispatching said one of said plurality of instruction segments, as modified by said modifying, to an execution unit of said processor.

38. (original)A method as described in Claim 37 wherein said instruction word is of a Very Long Instruction Word (VLIW) type.

39. (original)A method as described in Claim 38 further comprising advancing a position of said memory queue in response to a bit field of said trigger portion.

40. (original)A method as described in Claim 38 wherein said trigger portion and said one of said plurality of instruction segments are both specific to said execution unit.

41. (original)A method as described in Claim 40 wherein said execution unit is a memory execution unit.

42. (original)A method as described in Claim 40 wherein said execution unit is an arithmetic logic unit (ALU).

43. (original)A method as described in Claim 40 wherein said execution unit is a floating point unit (FPU).

44. (original)A method as described in Claim 40 wherein said execution unit is a branch unit.

45. (original)A method as described in Claim 40 wherein said identifying and said modifying are performed by microcode internal to said processor.